28.(new) The apparatus of claim 26, wherein the initialization module is further configured to reconfigure the driver to execute the error avoidance module upon generation of the FIFO interrupt.

## REMARKS

The Office Action mailed March 26, 2004 (Paper No. 7), has been received and reviewed. Claims 1-24 are in the case. Claims 1-24 stand rejected as being anticipated under 35 U.S.C §102(b) over Krakirian (U.S. Patent No. 5,450,546). By this amendment claims 3, and 6-8 have been cancelled without prejudice or disclaimer. Claims 25-28 have been added.

### REJECTION OF CLAIMS 1-24 UNDER 35 U.S.C. 102(b) IN VIEW OF KRAKIRIAN

Claims 1-24 stand rejected under 35 U.S.C. §102(b) as being anticipated by Krakirian. Applicant respectfully asserts that the defense of anticipation is improper. In order for a cited reference to anticipate, every element of the claimed invention must be identically disclosed in a single cited reference; and those elements must be arranged or connected together in a single reference in the same way as specified in the patent claim. Applicant asserts that the cited reference does not recite the claimed structure.

With respect to claims 1 and 25, Applicant asserts that Krakirian does not disclose a controller configured to generate a content limiting interrupt corresponding to a near capacity condition, as recited in newly amended claim 1. The device of Krakirian merely generates an SREQSTOP signal when the buffer is either completely full or empty. (Col. 8, lns. 40-58).

With respect to claim 2, applicant asserts that Krakirian does not "force" an error condition. The apparatus of Krakirian does not generate an error condition (e.g. a signal indicating that something has gone wrong), it is simply hardwired to generate a signals that stop the transfer of data.

(Col. 8, lns. 29-39). Krakirian does not teach the "forcing" of any error condition, the device of Krakirian simply acts according to its hardwired logic.

With respect to claim 4, applicant asserts that the microprocessor interface circuit 260 of Krakirian does not enable any executable data structures of an error avoidance module. The microprocessor interface circuit 260 of Krakirian merely initializes control registers to the number of blocks in buffer memory. (Col. 6 ln. 67 - col. 7 ln. 5). The number of blocks in a buffer memory is operational data, not executable data.

With respect to claims 5 and 27, the apparatus of Krakirian lacks an initialization module as recited in claim 4 upon which claim 5 depends. Furthermore, Figure 2 element 240 is hardwired to generate a SREQSTOP signal to the host interface circuit 270, therefore it cannot be said to be "enabled" to generate an interrupt by any other executable data structure. (Col. 8, lns. 40-50, Figure 3B-2 upper right corner - logic circuit for SREQSTOP signal).

With respect to claim 9-12, Applicant asserts that claims 9-12 are dependent on allowable claim 5 and are therefore allowable.

With respect to claims 13 and 26, Applicant asserts that Krakirian fails to disclose forcing an error condition corresponding to an unsuccessful transfer. The SREQSTOP signal of Krakirian signals to the CPU to stop transferring data, it does not in any way indicate to any other part of the apparatus that a transfer was unsuccessful. (Col. 8,lns. 29-58).

The disclosed invention forces an error condition corresponding to an overrun or underrun regardless of whether one actually occurred based on the error avoidance module determining that an undetected error could have occurred. In this manner, the disclosed invention compensates for time-gaps that would otherwise cause an error to be undetected.

The specification states that "in certain embodiments, the error condition may be forced if the value of the count is at least as large as the capacity of the buffer, accounting for the possibility that a buffer overrun occurred and went undetected." (Page 6, ln. 1-5). Accordingly, the disclosed invention, in some embodiments, forces an error condition corresponding to a buffer overrun regardless of whether an overrun occurred.

With respect to claim 14, Applicant asserts that claim 14 is dependent on allowable claim 13 and is therefore allowable.

With respect to claim 15, Applicant asserts that Krakirian does not disclose a device forcing an error condition corresponding to an unsuccessful transfer of bytes between devices after having transferred the bytes. The apparatus of Krakirian merely sends a signal that stops the transfer of bytes when a buffer is full. (Col. 8,lns. 29-58). It does not provide for an error signal corresponding to an unsuccessful transfer after the transfer has taken place. Thus, the device of Krakirian does not provide a method for correcting undetected errors as does the device of applicant.

With respect to claim 16, Applicant asserts that Krakirian does not teach the step of dynamically creating a content limiting interrupt. The device of Krakirian is hardwired to send the SREQSTOP signal. (Col. 8,lns. 29-58; Figure 3B-2 upper right corner - logic circuit for SREQSTOP signal). The logic controlling the SREQSTOP signal therefore could not be deemed to be dynamically created.

With respect to claim 17-21, Applicant asserts that claims 17-21 are dependent on allowable claim 16 and are therefore allowable.

With respect to claim 22, Applicant asserts that Krakirian fails to disclose an article configured to force an error condition corresponding to an unsuccessful transfer. The SREQSTOP

signal of Krakirian signals to the CPU to stop transferring data, it does not in any way indicate to any other part of the apparatus that a transfer was unsuccessful. (Col. 8,lns. 29-58).

With respect to claims 23 and 24, Applicant asserts that Krakirian fails to disclose an initializaiton module configured to modify executable data structured comprising a controller driver in order to enable an interrupt. The microprocessor interface circuit 260 of Krakirian does not enable any executable data structures of an error avoidance module. The microprocessor interface circuit 260 of Krakirian merely initializes control registers to the number of blocks in buffer memory. (Col. 6 ln. 67 - col. 7 ln. 5). The number of blocks in a buffer memory is operational data, not executable data.

With respect to claim 28, Applicant asserts that Krakirian fails to teach an initialization module configured to reconfigure a driver. The device of Krakirian is simply an electronic circuit and teaches no means to modify its operation to execute an error avoidance module.

In view of the foregoing remarks, Applicant believes the claims to be in condition for allowance. Reconsideration of claims 1, 2, 4, 5, and 9-28 in view of the above remarks is therefore respectfully requested. In the event the examiner finds any impediment to the allowance of the claims, Applicant respectfully requests that the examiner call the undersigned. Enclosed is Form PTO-2038 authorizing credit card payment in the amount of Forty-Three Dollars ($43.00) in payment of the addition of one independent claim.

DATED this 22nd day of June, 2004.

Respectfully submitted,

A. John Pate
Registration No. 36,234
Attorney for Applicant

Date: June 22, 2004

PATE PIERCE & BAIRD
550 Parkside Tower
215 South State Street
Salt Lake City, Utah 84111
Telephone: (801) 530-0330
Facsimile: (801) 530-5955